Presentation No. 212

SCA 2026 Supercomputing Asia
Gathering the Best of HPC in Asia

HPC Asia 2026

Adaptive MARL for Real-Time GPU Job Scheduling and Resource Utilization in HPC Systems

Vaibhav GP Mehta*, Masashi Kohda, Lam Le, M. Iwasa, H. Ito, and M. Nakamura

TAS Design Group Inc. and Morgenrot.Inc

TAS Design Group

MORGENROT

Keywords: Multi-Agent RL, Job Queue Optimization, Resource Allocation, AI/HPC Server, Real-Time Scheduling

Abstract

GPU-accelerated HPC clusters suffer from idle GPUs, long queues, and unfair slowdowns due to static, CPU-centric schedulers.

Approach:
We propose a real-time Multi-Agent Reinforcement Learning (MARL) scheduler that decomposes scheduling into job selection and GPU resource allocation, trained cooperatively using PPO.

Results:
Evaluated on 86,720 production Slurm jobs, our approach improves:

- GPU utilization by +11.8%
- Bounded slowdown by ~7%
- Sub-millisecond inference latency

Introduction

GPU-accelerated HPC workloads are heterogeneous and dynamic, while production schedulers rely on static heuristics such as FCFS and backfilling.

This mismatch leads to:

- Idle GPUs despite long queues
- Large jobs blocking short jobs
- Inefficient handling of bursty arrivals

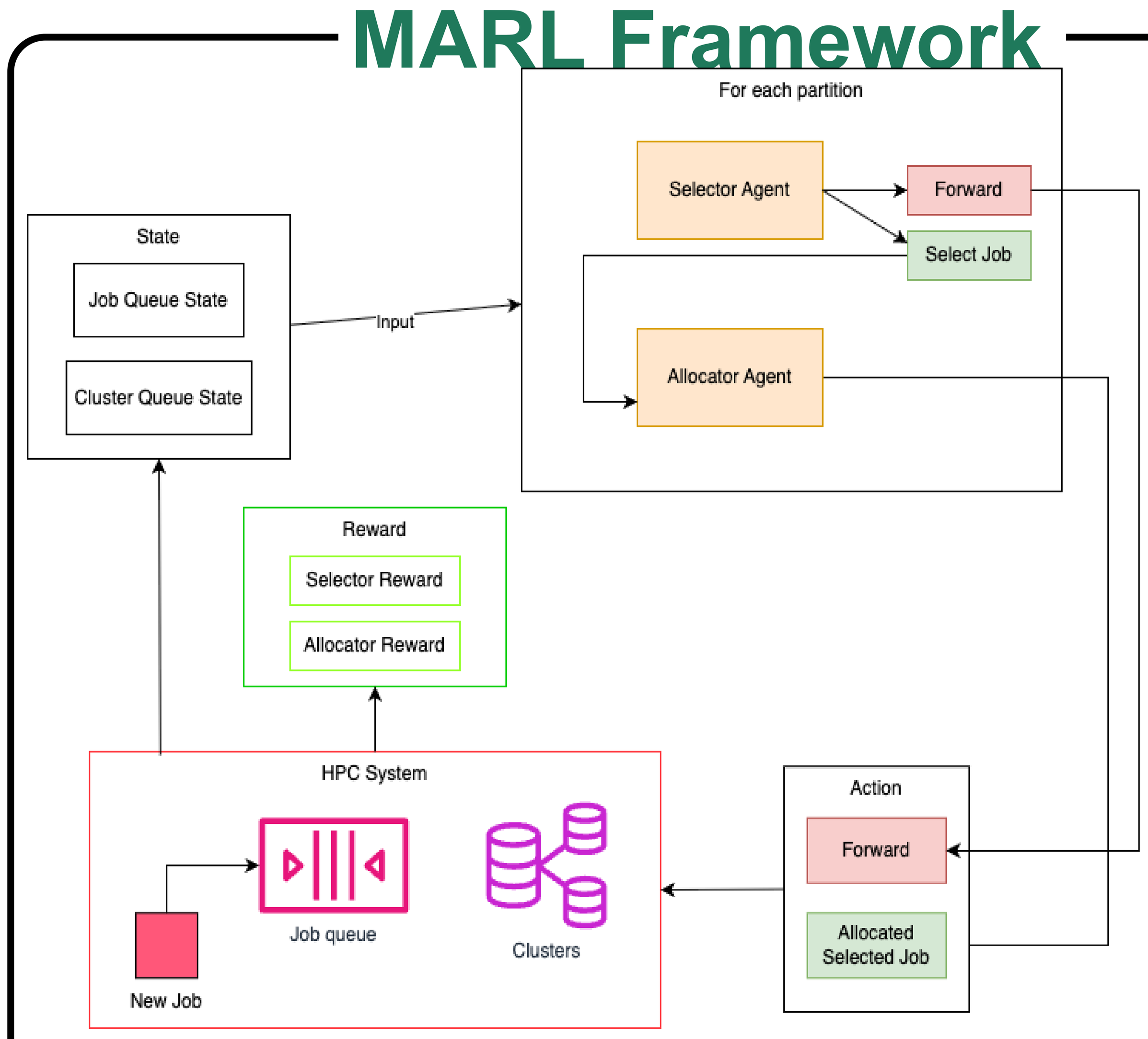
Key insight: Static heuristics cannot adapt to real-time workload variability in modern GPU clusters.

Problem Formulation

GPU scheduling in modern HPC systems involves two tightly coupled decisions under dynamic and heterogeneous workloads:

- **Job selection:** choosing which job to admit from a continuously changing queue
- **GPU allocation:** assigning heterogeneous GPU resources while avoiding fragmentation
- **Operational challenge:** static heuristics conflate these decisions, limiting adaptability under bursty arrivals and mixed job sizes
- **Observed impact:** idle GPUs, long waiting times, and degraded fairness

MARL Framework



Two-Agent Architecture

Selector Agent

- Chooses which job to admit from the queue
- Balances fairness and responsiveness

Allocator Agent

- Assigns GPUs/nodes to the selected job
- Accounts for hardware heterogeneity and fragmentation

Training

- Cooperative PPO
- Offline training on real Slurm traces
- Online inference in real time

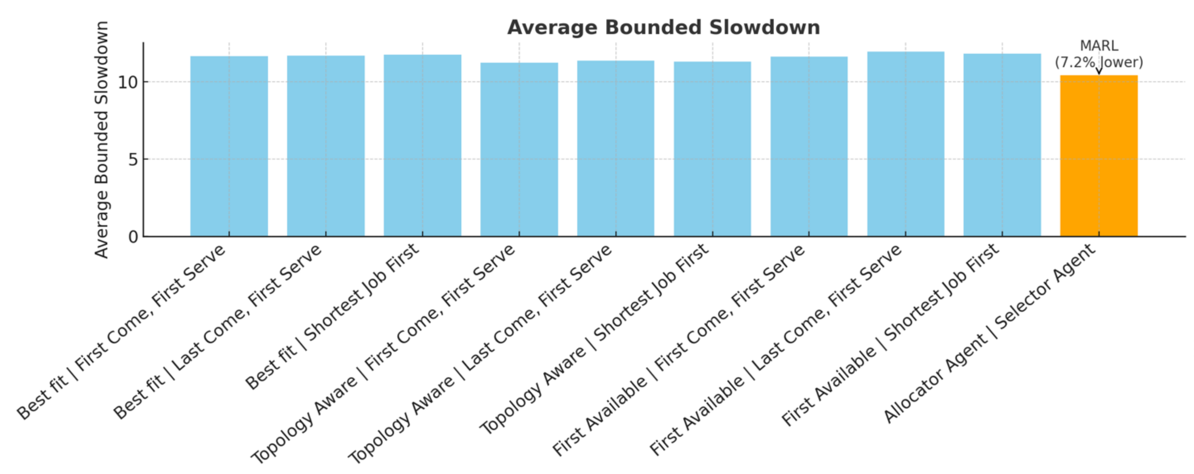
Optimizing Methodology

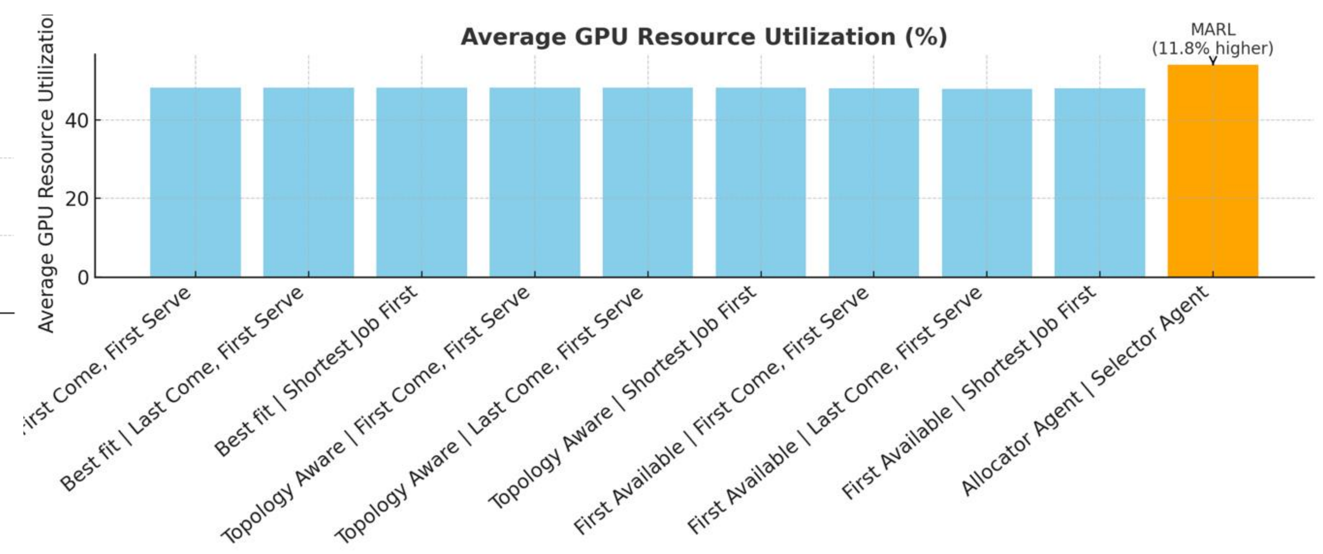
Realistic Evaluation:

- **Workload:** 86,720 production jobs (Spartan HPC cluster)
- **Cluster:** 30-node GPU partition
- **Baselines:** 9 allocator–selector combinations (Best Fit, Topology-Aware, First Available × FCFS, LCFS, SJF)

Metrics: Waiting time, Turnaround time, Bounded slowdown, GPU utilization

Key Outcomes





The scheduler jointly optimizes:

- ↓ Average waiting time
- ↓ Turnaround time

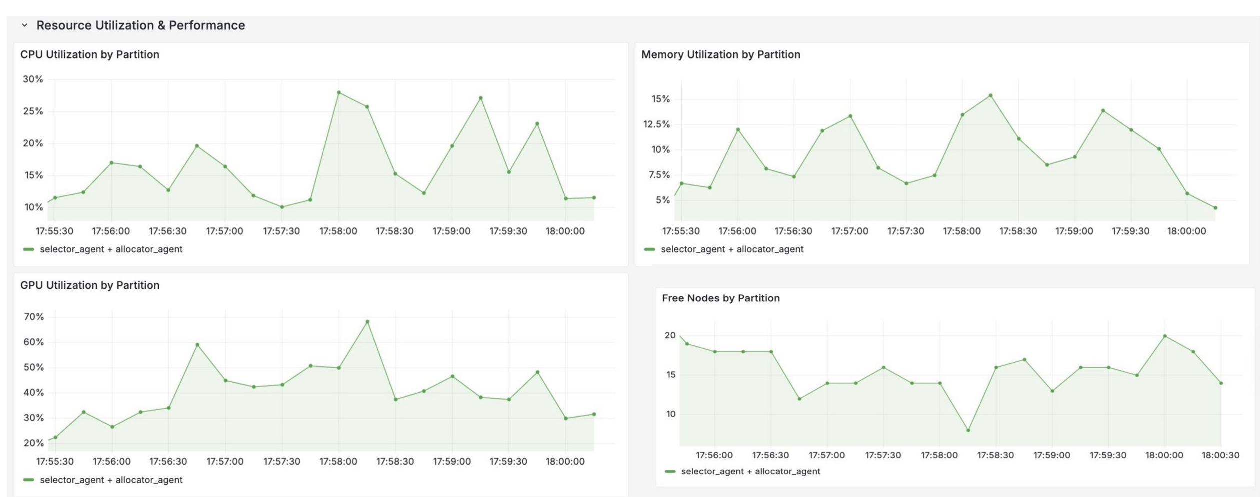
- ↓ ~7.2% reduction in bounded slowdown (fairness)
- ↑ +11.8% GPU utilization

Unlike heuristics, MARL adapts trade-offs dynamically as workload conditions change.

Production Readiness

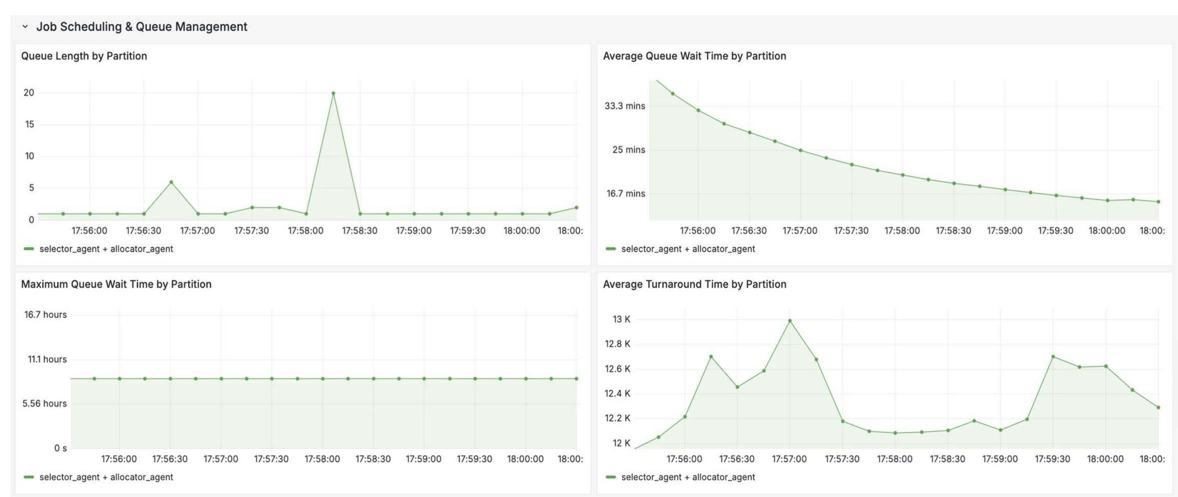
- **Real-time inference:** ~1 ms per scheduling decision
- **Scalable execution:** inference cost depends on model size, not cluster size
- **Safe training pipeline:** offline training with zero impact on live workloads
- **Native integration:** implemented as a lightweight Slurm plug-in
- **Workflow compatibility:** interoperates with existing HPC scheduling workflows

GPU Utilization by Partition



Sustained GPU utilization across partitions under live scheduling.
Demonstrates stable, high GPU usage with real-time multi-agent decisions.

Job Scheduling & Queue Management



Queue wait time decreases as agents adapt online.
Indicates faster job admission without sacrificing resource efficiency

Conclusion

Why This Matters?

- Reduces idle GPUs and wasted compute
- Improves fairness between short and long jobs
- Maintains responsiveness under dynamic workloads
- Deployable in real HPC systems today

Scheduler	Real Traces	Heterogeneous GPUs	Deployed	Multi-Agent
DeepRM	❌	❌	❌	❌
DL²	⚠️	⚠️	⚠️	❌
HRL (2025)	⚠️	⚠️	❌	❌
Ours	✅	✅	✅	✅

Decomposed MARL enables practical, fair, and real-time GPU scheduling for modern HPC clusters.

References

- Yahav Biran and Imry Kissos. 2025. Adaptive Orchestration for Large-Scale Inference on Heterogeneous Accelerator Systems Balancing Cost, Performance, and Resilience. (2025). https://doi.org/10.48550/ARXIV.2503.20074
- Jialin Cai, Hui Zeng, Feifei Liu, and Junming Chen. 2025. Intelligent Dynamic Multi-Dimensional Heterogeneous Resource Scheduling Optimization Strategy Based on Kubernetes. Mathematics 13, 8 (April 2025), 1342. https://doi.org/10.3390/math13081342
- Marco Canini, Ricardo Bianchini, Íñigo Goiri, Dejan Kostić, and Peter Pietzuch. 2025. Rethinking Cloud Abstractions for Tenant-Provider Cooperative Optimization of AI Workloads. (2025). https://doi.org/10.48550/arXiv.2501.09562
- Yongkang Dang, Minxian Xu, and Kejiang Ye. 2023. Resource Management for GPT-based Models Deployed on Clouds: Challenges, Solutions, and Future Directions. arXiv (January 2023). https://doi.org/10.48550/arxiv.2308.02970
- Q.L. Ding, Pengfei Zheng, Shreyas Kudari, Shivaram Venkataraman, and Zhao Zhang. 2023. Mirage: Towards Low-Interruption Services on Batch GPU Clusters with Reinforcement Learning. (November 2023). https://doi.org/10.1145/3581784.3607042